A PARALLEL MULTIBLOCK MESH MOVEMENT SCHEME FOR COMPLEX AEROELASTIC APPLICATIONS

Mark A. Potsdam*
Eloret Corp.
Moffett Field, CA

Guru P. Guruswamy†
NASA Ames Research Center
Moffett Field, CA

ABSTRACT

A scheme has been developed for the movement of multiblock, structured grids due to surface deformation arising from aeroelastics, control surface movement, or design optimization. Elements of the method include a blending of a surface spline approximation and nearest surface point movement for block boundaries. Transfinite interpolation is employed for volume grid deformation. The scheme is demonstrated on a range of simple and complex aeroelastic aircraft applications using Navier-Stokes computational fluid dynamics and modal structural analyses on parallel processors. Results are robust and accurate, requiring only minimal user input specification.

INTRODUCTION

Computational fluid dynamics (CFD) has reached a state where complex geometries can be analyzed in order to investigate complex flowfields and solve difficult engineering problems. CFD is also being used more frequently within multidisciplinary applications for design optimization, aeroelastics, control surface analysis and aeroservoelastics, and thermal analyses. While linear methods for these problems are well understood and relatively inexpensive, they present limitations. Non-linear methods are required to model complex flows, including vortex induced oscillations, transonic buffet, transonic flutter dip, and large control surface movements. One of the enabling technologies for multidisciplinary analyses using high fidelity methods has been the advent of large scale parallel computing platforms.

The combination of CFD with other disciplines frequently involves deforming geometries due to design modifications, surface movement, or structural loads. A scheme is required to deform the volume mesh to accept the surface movement. For structured grid algorithms, particular difficulties are posed at the grid interfaces in order to avoid repeated, expensive grid recoupling.

Numerous grid movement schemes exist for deforming block structured grids in multidisciplinary applications. The problem requires modifying a structured grid based on changes to a subset of its boundary points. The simplest method is to completely regenerate the grid based on the new surface [1,2], but this is only feasible for the simplest block structured geometries or zones and can be time consuming.

The most common methods generate perturbations on existing grids. Algebraic shearing is the predominant method used for perturbing single block grids [3,4] and has found use with overset grids as well [5,6]. In this method, surface point movement is projected along an index line emanating from the surface. The surface movement can be decayed to zero at the outer or overset boundary. Rotational movement can be added to maintain orthogonality at the surface. This scheme is simple to implement and is quite fast. However, it cannot be easily generalized to multiblock grids wherein the block face opposite the surface is not an outer boundary. In this case, the abutting blocks will influence the face movement. Additionally, large

---

* Senior Research Scientist/Engineer, AIAA Senior Member
† Senior Research Scientist, AIAA Associate Fellow

deflections can have adverse effects on grid quality.

Transfinite interpolation (TFI) is a more general three-dimensional method that is widely used [7,8,9,10] for cases involving multiple deforming boundary faces. TFI combines the speed and efficiency of an algebraic method with the ability to handle fully 3D perturbations. Modifications can be made to add robustness. Drawbacks relating to subfaces are examined in this paper with the aim of eliminating them to create a more general scheme.

More complex grid movement schemes involve a spring analogy [11,12,13]. Elasticity based schemes which include both linear and torsional springs have been shown to produce good quality meshes. However, they tend to be computationally and memory intensive and have found wider use on unstructured meshes where structured 3D interpolation methods are not as applicable.

All of these schemes make use of grid connectivity information. For multiblock grid interfaces, this can be problematic since points are not required to match at the interface. Even if a point happens to match on the boundary, the connectivity to other points can be different. This may result in different movements for points at the same location in space. Point-by-point schemes [14,15,16] remove this problem by individually moving grid points based on their position in space and their relation to the deforming surface. A drawback to this approach is that the lack of grid connectivity information can produce grids that are not smooth. Hartwich [14] presents a successful point-by-point scheme that is assisted by operating on grid interface subgrids. Taking advantage of the fact that the grids in use are typically multigridable, small subgrids (5x5), within which TFI is still performed, maintain a small amount of interconnectivity necessary for smoothness. Chen [16] uses a 3D boundary element method solver to both interface the fluid and structures surface grids and deform the fluids volume grid, assuring that the surface and volume grid distributions will be smooth and continuous with each other. A two zone approach is used to maintain grid quality near the surface. It remains to extend this methodology to general multiblock grids.

Objectives

The objective of this work is to develop a generalized multiblock moving grid scheme that removes some of the limitations present in current state-of-the-art methods.

The scheme used to deform the CFD grids must meet several requirements:

1) *Robustness*. The scheme must be robust enough to handle arbitrarily complex multiblock grids, including the Navier-Stokes clustering and singularities (i.e. axes, C-cuts) frequently found in CFD grids.
2) *Accuracy*. The grid deformation scheme must produce grids of acceptable quality to the flow solver. While positive Jacobians (volumes) are a minimum requirement, maintaining smoothness, orthogonality, and the overall quality of the original grid is required. Additionally, at the grid interfaces, point matched grids clearly must remain point matched. A desirable characteristic for abutted grids is not having to recalculate interpolation coefficients, necessitating that points in a region move in unison.
3) *Ease of Use*. Ideally the grid deformation scheme should be invisible to the user or require only minimal inputs. The user should not be burdened with describing any particular motion of the grid boundaries or corner points.
4) *Efficiency*. In a tightly coupled analysis, both the fluids and structures models advance every iteration. For dynamic, time accurate analyses, tight coupling is necessary. Therefore, the grid deformation scheme must be efficient since it may be required at every time step.
5) *Parallelizable*. The scheme must integrate with parallel computational codes and not result in excess communication costs or processor idle time.

The grid movement scheme developed here builds on and improves upon the grid movement implementation in ENSAERO [7] and the scheme by Hartwich [14]. Byun [7] uses transfinite interpolation (TFI) exclusively to move grid block corners, edges, faces, and interior points in order. While this scheme offers the advantages of speed

and efficiency, it also contains two major drawbacks:

1) *Subfaces*. TFI is a global methodology which interpolates multidirectionally based only on block boundary values. By itself TFI does not correctly handle subface grids, where two or more grids abut against one. This is due to the fact that, in general, for two adjoining grid faces **A** and **B** which abut against a single face **C**=**A**+**B**

$$\mathrm{TFI}(\mathit{face}(\mathbf{A})) + \mathrm{TFI}(\mathit{face}(\mathbf{B})) \neq \mathrm{TFI}(\mathit{face}(\mathbf{A}+\mathbf{B})) \quad (1)$$

because the adjoining **AB** boundary values are ignored in the **A**+**B** solution. A simple example of a 2D point matched interface where TFI breaks down without special treatment is shown in Figure 1, where the outer grid interfaces with 3 inner grids. The separation of the grid boundary faces after TFI is shown and results in the loss of point match.

2) *Input Specification*. User input to the scheme includes all grid block corner points and their motion relative to structural nodes. Additional input is necessary to handle specialized grid topologies. The extensive input can be time consuming and error prone. Requiring the user to specify the motion of the block boundaries relative to the structures can produce unexpected results.

In order to remove the subface limitation of the TFI scheme, coding logic could be written to determine subface connectivity in the point matched case and perform separate TFIs on the subsections. This has the advantage of being relatively inexpensive. The drawbacks include complicated coding and required access to interface blocking data. It is also not clear how general patched or chimera grid interfaces could be handled. For example, the patched grid shown in Figure 2, where a single grid abuts against numerous others in a complex manner, might be too difficult to handle correctly.

An alternate solution chosen in this work is to develop a point-by-point methodology that does not require grid connectivity information on the boundary faces. By basing face point movement on 1) distance to the nearest structural surface and 2) a surface spline approximation to the structural surface movement, the need for TFI on boundary faces or any connectivity information is removed.

METHODS

The application focus in this work is on aeroelastic analysis, the multidisciplinary coupling of aerodynamic and structural modeling using computational fluid dynamics (CFD) and computational structural dynamics (CSD). Due to the complex nature of many aeroelastic problems, extensive computational resources may be required. Throughout this work parallel processing strategies and parallel computers have been used to obtain efficient solutions.

CFD

The CFD code used in this work is ENSAERO (HiMAP) [17]. ENSAERO is a finite difference Navier-Stokes solver employing either an ARC3D central difference scheme or Goorjian-Obayashi upwinding. Recent improvements to the code include a multiblock scheme using abutting or overlapping grids, point matched or non-point matched at the interfaces. Both Baldwin-Lomax and Spalart-Allmaras turbulence models are available. The code has been parallelized to run on parallel supercomputers such as the SGI Origin 2000 or IBM SP2. Coarse grain parallelization is implemented such that each processor performs calculations on one or more grid blocks. Blocks are distributed automatically by the load balancing algorithm based solely on the number of points. Zonal interface data is communicated using MPI message passing. The code has been optimized for the cache memory architectures typically found on these machines, however, suitable performance on Cray vector machines is maintained.

CSD

A modal structural model [18] is coupled with the ENSAERO flow solver. The modal structural approximation is similar to a Rayleigh-Ritz method and uses a linear combination of mode shapes to approximate structural response to a load. Mode shapes may be determined using experimental data or a finite element model. One and three degree of freedom (DOF) per node models are supported using unstructured triangular meshes to model the structural geometry.

Fluid-Structure Interface

A fluid-structure interface is required to transfer surface loads from the CFD solution to the structural model and to transfer surface deflections from the CSD solution to the CFD model. In general, the CFD surface grid and structures mesh will not be similar, with each discipline focusing on different critical areas. A lumped load point-to-cell projection method is used in ENSAERO to link CFD grid points to modal nodes [19].

The CFD and structures codes are not fully integrated, but remain as separate executables. This allows easier integration of alternate analysis methods such as unstructured CFD or full finite element analysis. The modules are run concurrently under the MPIRUN multiple process manager and communicate using MPI message passing.

The structures module only deforms the CFD grid surfaces corresponding to the structural surface. The remaining piece to complete the fluid-structure interfacing is the deformation of the parts of the block faces not included in the structural surface (flow through interfaces, wakes, planes of symmetry, axes) and the interior volume points.

CFD Grid Block Face Deformation

A moving grid scheme has been developed that does not use TFI for block boundary faces. This means that the movement of a face point does not depend on the movement of neighboring points, a condition necessary to remove dependency on grid topology (subfaces). This condition is satisfied if the face point movement is based on location in space (x,y,z) and distance to the deforming surface [14]. The scheme combines several elements to compute face point movement.

1) *Nearest Surface Point*. Points near a deforming surface follow the movement of the nearest surface point.
2) *Surface Spline*. Away from the deforming surface, points move smoothly based on a surface spline approximation of the deforming surface movement.
3) *Blending Function*. A blending function smoothly overlaps the (1) near- and (2) far-surface movement regions.
4) *Decay Function*. A decay function is used to decrease point movement away from the body.

Nearest Surface Point

The most logical means of computing the movement of a block face point is to base it on the movement of the nearest structural surface point. This creates a shearing movement for points associated with a surface point. It is a relatively good strategy for treating tightly clustered Navier-Stokes grids and for avoiding local grid line crossing in the near-body region. While not included here, modifications can be made to maintain grid orthogonality at the surface by accounting for surface point rotation in addition to translation [5].

The nearest point strategy, however, may not result in a smooth deformation of the grid. Points adjacent to each other may slave off different surface points with different deformations, resulting in a jagged appearance of the grid. To improve smoothness, the nearest surface point should be an interpolated point on the structured surface as determined by the shortest perpendicular distance, not necessarily a grid point.

A further difficulty arises when neighboring grid points do not reference surface points in close proximity to each other. An example of a supercritical airfoil is shown in Figure 3. The airfoil has been rotated about the leading edge, and field points follow the movement of only the nearest interpolated surface point. Two points near the center of a circular arc approximating the lower surface cove are highlighted. They slave off surface points which are not in close proximity, and mesh discontinuity results. Similar topologies occur in internal ducts. In such regions an alternate, smoothly varying grid movement scheme is required.

Surface Spline

A surface spline [20] is a mathematical equation for interpolating a multivariate function. It contains structural behavior by basing its derivation on the small deflection equation of an infinite plate. Modifications for scaling, symmetry, and smoothing are employed. Using a set of arbitrarily placed points with defined displacements, functions are devised to compute

displacements (Δx,Δy,Δz) everywhere as a function of location in space (x,y,z). for example,

$$\Delta z_{spline} = f(x,y,z) \tag{2}$$

This meets the same conditions as the nearest surface point strategy, except that the function is smooth and differentiable. Its accuracy, however, is dependent on the number of points used to generate the function.

Surface spline generation requires the solution of a system of linear equations, and its evaluation contains natural logarithms. Currently, subsets of the fluid-structure interface points are specified to be used in the spline. The number of input points should be limited from the complete fluids and/or structures point sets to keep the cost of the surface spline evaluation reasonable. While previous works have frequently employed the surface spline as a method for fluid-structure interfacing [21], it is used here in a novel manner to compute the movement of the off-body block boundary face points.

The lumped load point-to-cell methodology is retained for transferring deflections from the structural mesh to the CFD surface grid. Because the spline may lack the fidelity and accuracy of the actual surface movement, it cannot be used confidently in the Navier-Stokes clustered regions near the surface. Here it is replaced by the nearest surface point movement strategy. The surface spline is employed where the nearest point movement scheme may fail due to having "closest" points not in close proximity or in areas where high fidelity is not a concern.

Blending Function

A blending function is required to smoothly transition between the nearest (interpolated) surface point movement and the surface spline approximation methods. Because the nearest point movement fails when the field of view of a point becomes too large, a reasonable flag for blending can be based on a ratio between the distance to the nearest surface grid point and the distance to the next nearest distinct surface grid point. Near the surface the inverse of this ratio, $\bar{d}$, approaches infinity as the nearest distance approaches zero and/or the next nearest is large in comparison. In the field away from the surfaces $\bar{d}$ will be order 1. The blending weight factor, determined by experimentation, is defined as

$$\begin{aligned} wt_{corr} &= 1 - \exp[-(\bar{d}^2 - 1)^m] \\ \bar{d} &= \frac{d_{next\_nearest\_point}}{d_{nearest\_point}} \\ m &= 0.5 \text{ - } 2.0 \end{aligned} \tag{3}$$

This function is still under development.

Near the surface the weighting function is one, decaying to zero in the field. It is applied to a correction term for the surface point movement, $\Delta\tilde{z}_{corr}$, which is the difference between the actual interpolated surface point movement, $\Delta\tilde{z}_{struct}$, as calculated by the structural model and the interpolated surface point movement as predicted by the surface spline, Δz_{spline},

$$\Delta\tilde{z}_{corr} = \Delta\tilde{z}_{struct} - \Delta\tilde{z}_{spline} \tag{4}$$

The tilde signifies values computed at the nearest interpolated surface point.

Decay Function

To keep the movement of the grid away from the body to a minimum, changes are decayed using an overall weighting function,

$$\begin{aligned} wt_d &= \exp\left[-\beta\bar{r}^2\right] \\ \bar{r} &= \frac{d_{nearest\_point}}{d_{ref}} \end{aligned} \tag{5}$$

The decay function is based on distance to the nearest surface and, with β = 1/4, decays to essentially zero at $d_{nearest\_point} = 4d_{ref}$. However, there is no requirement for using the decay function as there are no difficulties associated with movement of the outer boundaries which can be included in the surface spline definition.

Grid Block Face Point Displacement

The final displacement of the grid boundary face point can be calculated as,

$$\Delta z(x,y,z) = wt_d \cdot \left[\Delta z_{spline}(x,y,z) + wt_{corr}\Delta\tilde{z}_{corr}\right] \tag{6}$$

Similar equations can be written for Δx and Δy when using a 3D structural model.

CFD Volume Grid Deformation

The structural model and the grid movement scheme outlined above define the movement of the six block faces. Although the scheme would be well suited for deformation of interior grid points, in particular chimera grid boundaries, fringe points, and holes, it is more costly in comparison with algebraic methods. Transfinite interpolation [22] is used to compute the interior volume deformation due to its low cost and efficiency. The mesh movement increment ($\Delta x,\Delta y,\Delta z$) is interpolated, not the actual point locations (x,y,z). For poor initial quality grids, modification to the TFI blending functions may be required. Typical TFI schemes use blending functions that are linear in arc length, *S*. Exponential blending functions [23] in the non-viscous directions work well for regions of poor grid quality,

$$\alpha_1(S) = (1-S)\cdot e^{-KS}$$
$$\alpha_2(S) = S\cdot e^{-K(1-S)} \qquad (7)$$

where K = 0 reverts to the linear case, and K = 2.5 is used for increased robustness of the TFI scheme.

Implementation

The face movement scheme is applied to all points on the block faces not prescribed by structural motion. Additionally, it is applied to interior points that are donors for overlapped interfaces. Because of the inherent properties of the scheme, point matched points are guaranteed to have the same movement regardless of their connectivity in their respective block topologies. Non-point matched regions will move together in a smooth manner. Plane of symmetry points can be flagged in order to zero out their movement in the symmetry direction. Although the spline and structural models maintain symmetry, the nearest surface point to a symmetry plane point may not lie on the plane of symmetry and may result in movement in the symmetry direction.

The nearest and next nearest point distances can be calculated in a preprocessing step. The use of an efficient alternating digital tree (ADT) [24] significantly speeds this process. A further enhancement to the ADT is to begin with a reasonable starting guess and only back up in the tree as far as necessary, rather than starting at the root for each point. This modification works especially well for face points that are tightly clustered and for next nearest neighbor searches. Once a nearest surface point is found, searching the surrounding quadrilaterals determines the nearest interpolated surface point and its interpolation coefficients.

In a parallel environment, each processor performs the deformation of its own block boundaries and volumes. The only global data needed is the movement of the complete structural surface. The structures module broadcasts this to all fluids modules using MPI and MPIRUN. With this data available, each fluids module is able to independently generate the surface spline and deform the boundaries of its blocks. No grid interface information need be communicated between blocks.

User input to the scheme is fairly simple. The main input is the CFD surface definition specifying the structural surface, data required for any fluid-structure interaction problem. Surface patches can be specified as structurally deforming, fixed, or undergoing rigid body motion. Additionally, the user is required to specify a subset of each surface patch for use in defining the surface spline. The input consists of the numbers N_i and N_j for each patch where N_ixN_j equal arc length spaced points on the four boundary edges are automatically generated. For edges that undergo significant deformation in the edge direction, higher resolution by the spline is required, and more points in that direction are used in the subset. For example, a wing patch may require several spline points in the spanwise direction but only a few in the chordwise direction, which undergoes smaller deflections. Surface spline subset points for a wing/fuselage geometry are shown in Figure 4. Also shown are the structural surface patch boundaries which coincide with grid block boundaries. Points on the fuselage are fixed. It may not be necessary to subset both the upper and lower wing surfaces, and some surface patches need not contribute to the spline. A quick check of surface deflection when sample deformations are applied can assist in determining appropriate surface spline points. The number of surface spline points used results in a trade-off between cost and grid quality.

RESULTS

The current mesh movement scheme was implemented in ENSAERO. Several applications are shown to illustrate the flexibility of the method

on a range of multiblock configurations, from a 2D airfoil to a full aircraft configuration. All CFD Navier-Stokes grids are clustered for a y^+ of order one at the viscous surface. Convergence of steady state results was determined by force/moment output and wing pressures. CFD solutions use the upwind option.

Two-Dimensional Airfoil

A single block two-dimensional airfoil test case is shown in Figure 5 in its original and deformed position. The C-mesh airfoil grid with open trailing edge and wake has been translated and rotated from an initial zero degrees angle of attack position. Original grid quality and smoothness are maintained throughout a large range of motion. No deficiencies are seen where the grid motion transitions between nearest surface point movement and the surface spline approximation methods.

Arrow Wing-Body

Aeroelastic results for a low aspect ratio Boeing arrow wing-body (BAWB) configuration were calculated at transonic conditions ($M = 0.85$, $\alpha = 7.93°$) using the Baldwin-Lomax turbulence model with Degani-Schiff cut-off. Figure 6 shows a comparison of rigid and aeroelastic wing pressures with experimental data. It should be noted that the 1 DOF mode shapes are artificial. In actuality, the model is quite stiff, as indicated by the good agreement between the test data and the rigid solution. The CFD solutions capture the leading edge vortex equally well, indicating sufficient quality in the deformed grid has been maintained with no added discrepancies at the grid block interfaces. The grid contains 1.2 million points (2x 155 streamwise x 67 spanwise x 59 normal). The spline points shown in Figure 6 are chosen to adequately model expected aeroelastic motion, with clustering in regions of higher deflection towards the wing tip. In Figure 7 wing pressures and wing tip trailing edge deflections using the current grid movement scheme are compared with aeroelastic results calculated using the full TFI scheme [7]. Results are on a coarse grid (2x 110x58x40) but are in good agreement with each other.

The BAWB grids are H-H topology, abutted ahead of and behind the leading and trailing edges in the normal (boundary layer) direction. Splits in both the streamwise and spanwise direction result in 8 blocks. The original and deformed faces of a streamwise block interface plane in the midchord region are shown in Figure 8 for the coarse grid. Both the interface and a wing tip close-up are seen to have deformed smoothly. The coarse grid also serves to illustrate the robustness of the modified TFI scheme with the exponential blending functions. Regions in the grid with poor grid quality are deformed without further degradation (Figure 8 inset right). For the full TFI scheme, it was previously necessary to move the Navier-Stokes near-body region by shearing. Coding logic was required to identify and extract the viscous region which is now handled seamlessly with one methodology.

The coarse grid solution was run on 9 processors of an SGI Origin 2000 – one block per processor and one processor for the structures module. A breakdown of the time spent deforming the grids as a percentage of the total time per iteration on each fluids processor is shown in Table 1 in the BAWB column. The grid deformation scheme requires 26% of the total time per iteration. The spline uses 87 points, and its evaluation cost labeled 'Deform boundaries' is seen to be the largest. Spline generation time is minimal, and the algebraic TFI volume deformation is quite efficient.

DAST ARW-2 Wing/Fuselage

A high aspect ratio high wing Aeroelastic Research Wing (ARW-2) was tested extensively in the NASA LaRC Transonic Dynamics Tunnel under the Drones for Aerodynamic and Structural Testing (DAST) Program [25]. The model was built to replicate full-scale wing structure and is realistically flexible. Rigid and aeroelastic ENSAERO Navier-Stokes calculations using the Spalart-Allmaras turbulence model were performed at transonic buffet conditions ($M = 0.8$, $\alpha = 3.0°$) with air as the test gas. Wing pressure comparisons in Figure 9 show rigid (jig), modal and beam structural models, and experimental data. Good agreement is seen between the two grid deformation/aeroelastic schemes. The modal structural model uses the current grid movement scheme on a 1.7 million point multizone grid. The first five mode shapes (1 DOF) are included. A single block wing/fuselage C-O grid (289 streamwise x 86 spanwise x 67 normal) was split resulting in 10 blocks. The beam model calculations were performed on a single zone C-H grid (289 streamwise x 76 spanwise x 70 normal)

using the wing version of ENSAERO. Each spanwise constant Y grid plane independently undergoes rigid body translation and rotation in order to model the aeroelastic motion. The beam results do not include a fuselage, which has been determined to have minimal effect on the outer portion of the wing.

Aeroelastic wing pressure comparisons with experimental data are reasonable given the significant amount of separation present at this condition. Calculated wing tip deflection at the rear spar is 4.83 inches up compared with the experimental deflection of 5.18 inches. The model wing semispan is 113.92 inches.

Results used 11 processors of an SGI Origin 2000. A timing breakdown as a percentage of a flow solver step is shown in Table 1. The spline uses 109 points (Figure 4). The performance of the ARW computations are consistent with BAWB figures.

Table 1. Timing Breakdown for Grid Deformation

Operation	Time (% of flow solver step)		
	BAWB (N-S)	ARW (N-S)	L1011 (Euler)
Generate spline	1.7	0.6	2.4
Deform boundaries	19.0	17.6	25.9
Deform volume	5.7	6.3	6.2
Total grid deformation	26.4	24.5	34.5

Lockheed L-1011

A final test case is a full configuration Lockheed L-1011-500 transport aircraft [26]. The configuration represents a .02 scale wind tunnel model which was used extensively for flutter testing. The model includes the wing, fuselage, underwing nacelle, pylon and core cowl, vertical and horizontal tails, and sting. The aft engine is not present. Inviscid rigid and forced motion aeroelastic analyses were performed at Mach 0.88 and $\alpha = 0.0°$. The grid system contains 36 blocks and 9 million points. Although the surface normal spacings are appropriate for inviscid analyses, the grids were generated with the eventual aim of Navier-Stokes flutter calculations. For viscous analysis the points would be reclustered towards the surfaces, but the number of points would not increase. All block interfaces are point matched. Of particular note, the grid system is quite complicated, containing numerous degeneracies and blocks with multiple solid surface faces. The complex multiblock grid structure in the nacelle/pylon region is shown in Figure 10. Five structural modes are used: 1) first wing bending, 2) nacelle/pylon vertical, 3) nacelle/pylon lateral, 4) second wing bending, and 5) first wing torsion. They include significant interaction between the wing and the nacelle/pylon. The fuselage, sting, and empennage are rigid. The first 4 mode shapes (3 DOF) imposed on the unstructured modal mesh are shown in Figure 11.

Forced motion calculations using mode shapes 1 and 3 are shown in Figure 12. Test deflections were also performed with all modes active. This confirms the ability of the scheme to perturb the volume grids based on complex surface movement of an arbitrary multiblock configuration. No negative Jacobians are produced, and point matched block interfaces are maintained.

For the L-1011 the CFD flow solver runs most efficiently on 28 processors of an SGI Origin 2000. Timing breakdowns for the grid motion are shown in Table 1. The spline uses 214 points (Figure 10). Due to the complexity of the grids, it was necessary to use additional spline points in the wing/pylon junction region so that the spline more accurately modeled the motion. The increased percentage cost of the grid deformation scheme is due to the inviscid analysis. Including viscous terms would increase the cost per step and reduce the percentage costs of grid deformation to be more in line with the other viscous test cases. Additionally, since the boundary deformation scheme scales with the number of boundary points and the flow solver scales with the number of volume points, improved efficiency on larger numbers of processors could have been gained by considering both quantities in the load balancing.

CONCLUSIONS

A grid movement scheme has been developed for general multiblock configurations. It includes improvements over previous schemes that have limitations on grid block interface topology and robustness in Navier-Stokes clustered regions. The current method handles

point matched and non-point matched, overlapped and abutted interfaces. This is accomplished by basing block face point movement only on location of the point in space using a combination of nearest surface point and surface spline approximation. Volume grid movement is performed efficiently using transfinite interpolation. The scheme is applied effectively to a range of complex, multiblock aeroelastic configurations. A computational penalty is paid for generality and ease of use but can be reduced by judicious inputs.

ACKNOWLEDGMENTS

The work of the first author was completed under NASA Ames Fluid Dynamics Analysis contract NAS2-14109. Calculations were performed using the resources of the High Performance Computing and Communication Program (HPCCP) and Numerical Aerodynamic Simulation (NAS) facility at the NASA Ames Research Center. Continued support from Cathy Schulbach, program manager of the HPCCP Computational Aerosciences (CAS) Program, is gratefully acknowledged.

REFERENCES

1. Farhangnia, M. Guruswamy, G. P., and Biringen, S., "Transonic Buffet Associated Aeroelasticity of a Supercritical Wing," AIAA Paper 96-0286, January 1996.

2. Slater, J. W., "A Moving Grid Capability for NPARC," AIAA Paper 98-0955, January 1998.

3. Steger, J. L. and Bailey, E., "Calculation of Transonic Aileron Buzz," *AIAA Journal*, Vol. 18, No. 3, 1980, pp. 249-255.

4. Campbell, R. L. and Smith, L. A., "A Hybrid Algorithm for Transonic Airfoil and Wing Design," AIAA Paper 87-2552, August 1987.

5. Morton, S. A., Melville, R. B., and Visbal, M. R., "Accuracy and Coupling Issues of Aeroelastic Navier-Stokes Solutions on Deforming Meshes," *Journal of Aircraft*, Vol. 25, No. 5, September-October 1999, pp. 798 –805.

6. Gee, K. and Rizk, Y. M. "OVERAERO-MPI: Parallel Overset Aeroelasticity Code," HPCCP/CAS Workshop 1998, August 24-26 1998, NASA CP-1999-208757, January 1999.

7. Byun, C., and Guruswamy, G. P., "A Parallel, Multi-block, Moving Grid Method for Aeroelastic Applications on Full Aircraft," AIAA Paper 98-4782, September 1998.

8. Reuther, J., Jameson, A., Farmer, J., Martinelli, L., and Saunders, D., "Aerodynamics Shape Optimization of Complex Aircraft Configurations via an Adjoint Formulation," AIAA Paper 96-0094, January 1996.

9. Wang, Z. J. and Przekwas, A. J. "Unsteady Flow Computation Using Moving Grid with Mesh Enrichment," AIAA Paper 94-0285, January 1994.

10. Jones, W. T. and Samareh-Abolhassani, J., "A Grid Generation System for Multi-Disciplinary Design Optimization," AIAA Paper 95-1689, June 1995.

11. Batina, J. T., "Unsteady Euler Algorithm with Unstructured Dynamic Mesh for Complex-Aircraft Aeroelastic Analysis," AIAA Paper 89-1189.

12. Bartels, R. E., "An Elasticity-based Mesh Scheme Applied to the Computation of Unsteady Three-Dimensional Spoiler and Aeroelastic Problems," AIAA Paper 99-3301, June 1999.

13. Farhat, C., Degand, B., Koobus, B., and Lesoinne, M., "An Improved Method of Spring Analogy for Dynamic Unstructured Fluid Meshes," AIAA 98-2070, April 1998.

14. Hartwich, P. M., and Agrawal, S., "Methods for Perturbing Multiblock Patched Grids in Aeroelastic and Design Optimization Applications," AIAA Paper 97-2038, June 1997.

15. Anderson, W. K. and Venkatakrishnan, V., "Aerodynamic Optimization on Unstructured Grids with a Continuous Adjoint

Formulation," AIAA Paper 97-0643, January 1997.

16. Chen, P. C. and Hill, L. R., "A Three-Dimensional Boundary Element Method for CFD/CSD Grid Interfacing," AIAA Paper 99-1213, April 1999.

17. Guruswamy, G. P., Byun, C., Farhangnia, M., and Potsdam, M. A., "User's Manual for HiMAP: A Portable Super Modular 3-Level Parallel Multidisciplinary Analysis Process," NASA TM-1999-209578, September 1999.

18. Guruswamy, G. P., "Unsteady Aerodynamic and Aeroelastic Calculations for Wings Using Euler Equations," *AIAA Journal*, Vol. 28, No. 9, March 1990, pp. 461-469.

19. Guruswamy, G. P. and Byun, C., "Fluid-Structural Interactions Using Navier-Stokes Flow Equations Coupled with Shell Finite Element Structures," AIAA Paper 93-3087, July 1993.

20. Harder, H. L., and Desmarais, R. N., "Interpolation Using Surface Splines," *Journal of Aircraft*, Vol. 9, No. 2, February 1972, pp. 189-191.

21. Smith, M. J., Hodges, D. H., and Cesnik, C. E. S., "An Evaluation of Computational Algorithms to Interface Between CFD and CSD Methodologies," Report WL-TR-96-3055, Nov. 1995.

22. Thompson, J. F., Warzi, Z. U. A., and Mastin, C. W., *Numerical Grid Generation, Foundations and Applications*, Elsevier Science Publishing Company, New York, NY, 1985.

23. Eriksson, L.-E., "Practical Three-Dimensional Mesh Generation Using Transfinite Interpolation," *SIAM Journal of Sci. Stat. Comput.*, Vol. 6, No. 3, July 1985, pp. 712-741.

24. Aftosmis, M., "Lecture Notes on 'Solution Adaptive Cartesian Grid Methods,'" Von Karman Institute for Fluid Dynamics, Lecture Series 1997-02, March 1997.

25. Seidel, D. A, Eckstrom, C. V., and Sandford, M. C., "Transonic Region of High Dynamic Response Encountered on an Elastic Supercritical Wing," *Journal of Aircraft*, Vol. 26, No. 9, September 1989, pp. 870-875.

26. Goodwin, S.A., Byun, C., and Farhangnia, M., "Aeroelastic Computations Using A Parallel Computing System," AIAA Paper 99-0795, January 1999.

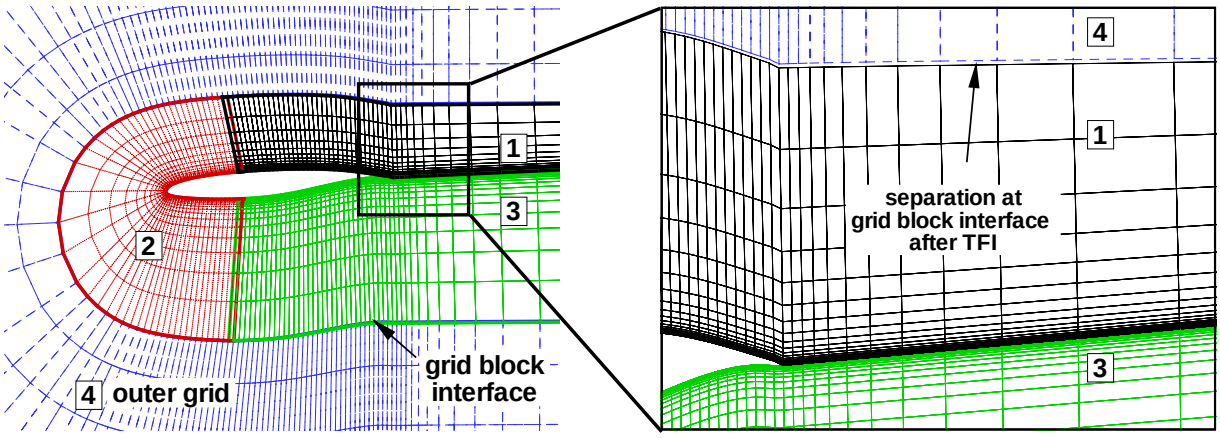

Figure 1 TFI Grid Interface Separation with Subfaces

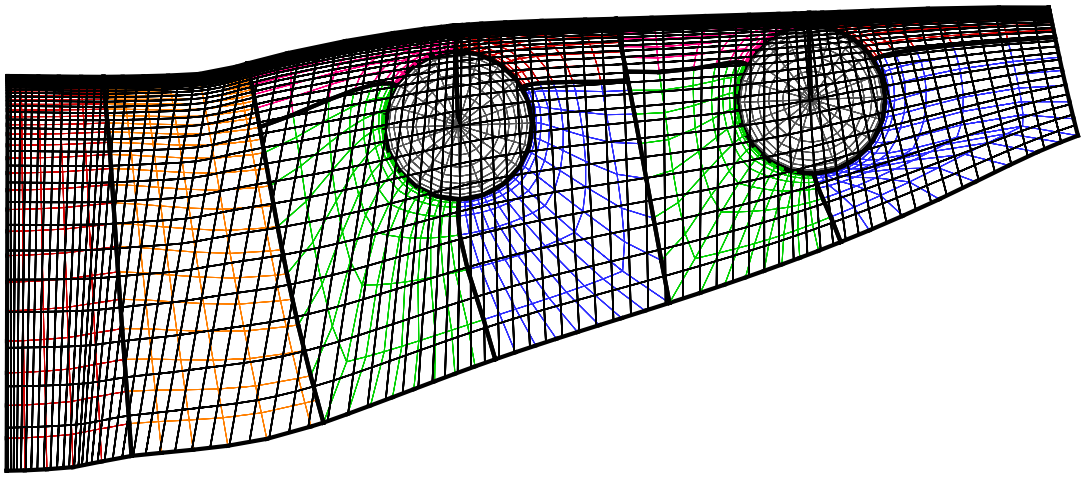

Figure 2 Abutted Non-Point Matched Subfaces in Complex Multiblock Configurations

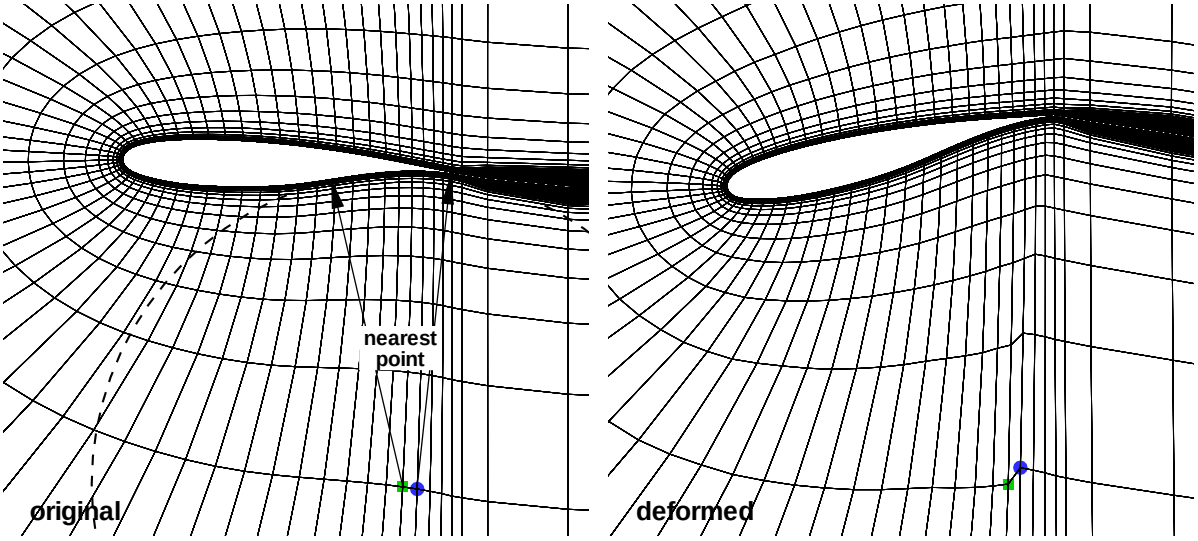

Figure 3 Errors in Grid Deformation Based on Nearest Surface Point Movement

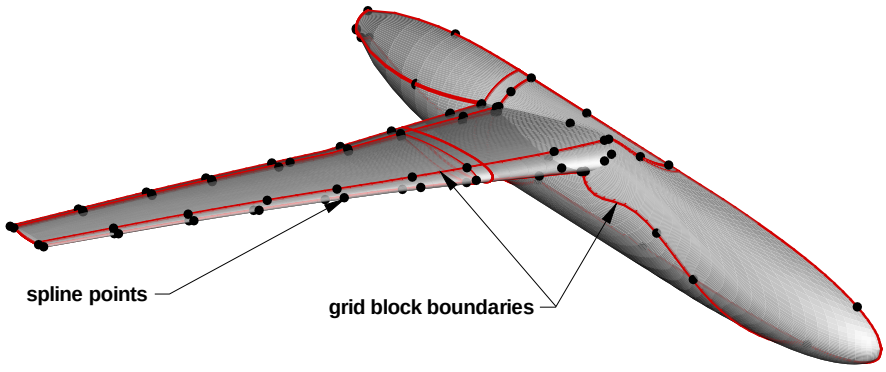

Figure 4 Point Subset for Surface Spline

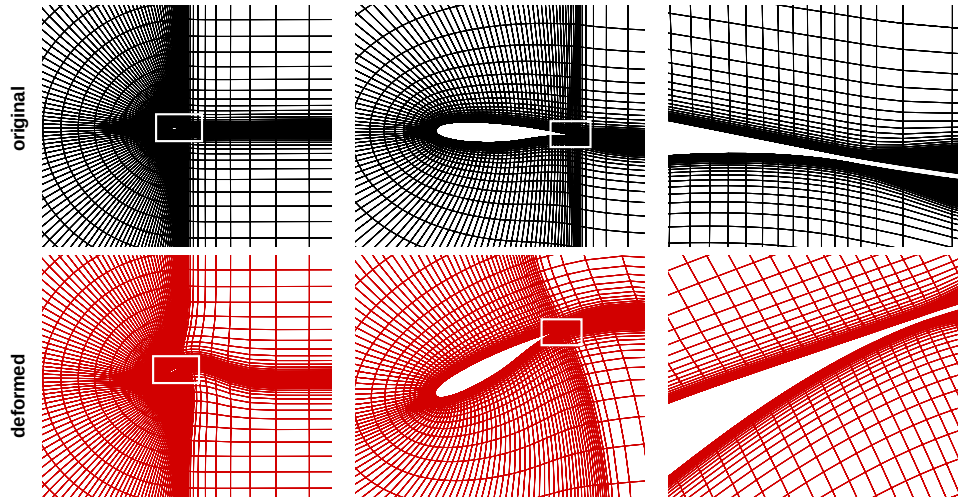

Figure 5 Deformation of 2D Airfoil C-mesh

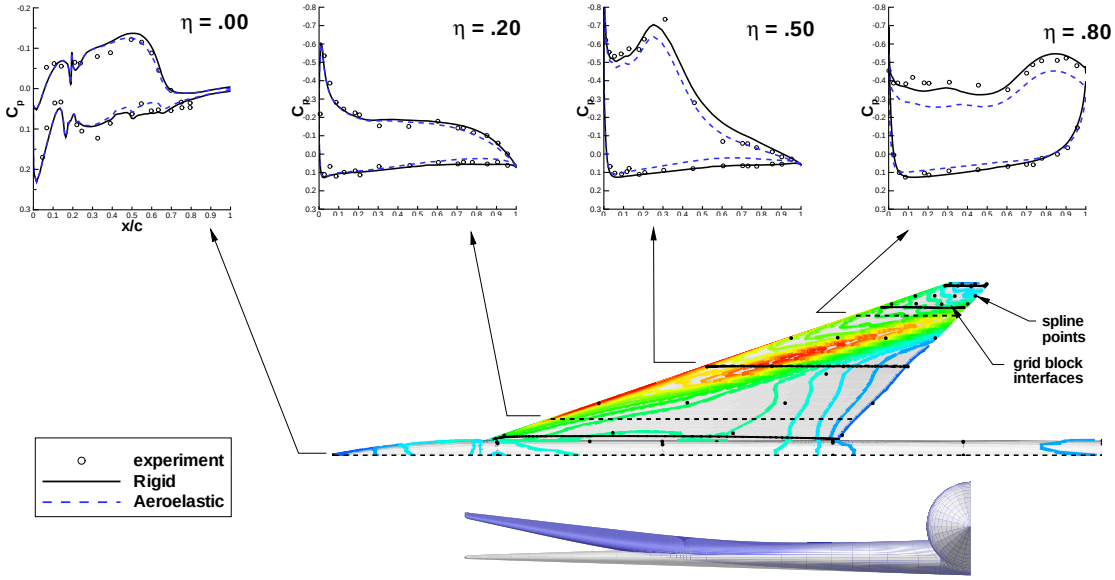

Figure 6 Arrow Wing-Body Navier-Stokes Aeroelastic Solution, M = 0.85, α = 7.93°, Re = 9.4 million, Baldwin-Lomax, Fine Grid

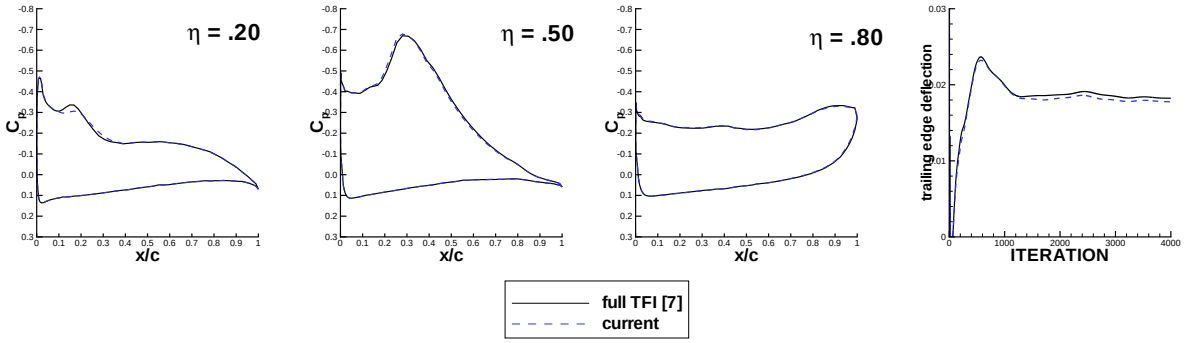

Figure 7 Arrow Wing-Body Comparison of Grid Deformation Schemes, Wing Pressures and Wing Tip Trailing Edge Deflection, Coarse Grid

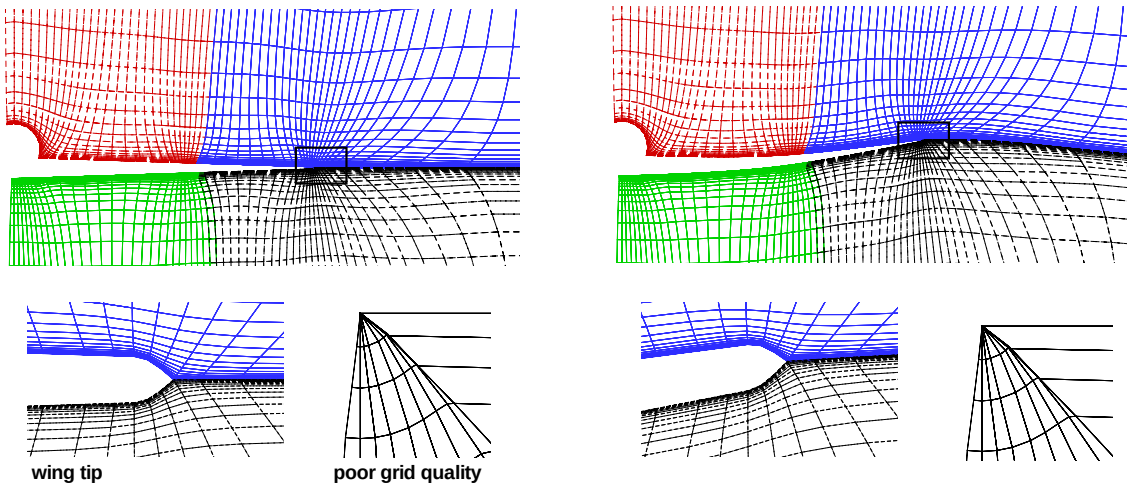

Figure 8 Arrow-Wing Body Grid Block Face Movement, Wing Tip Close-up and Poor Grid Quality

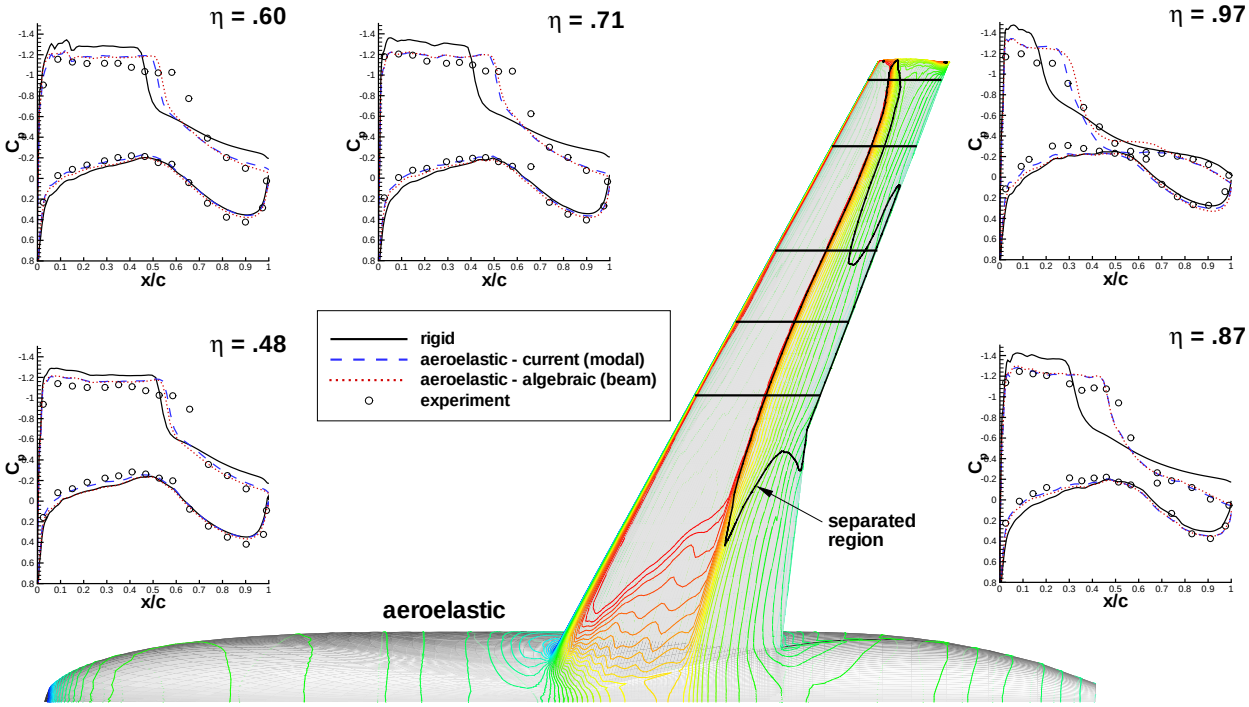

Figure 9 DAST ARW Navier-Stokes Aeroelastic Solution, M = 0.80, α = 2.98°, Re = 1.3 million, q = .72 psi, Spalart-Allmaras

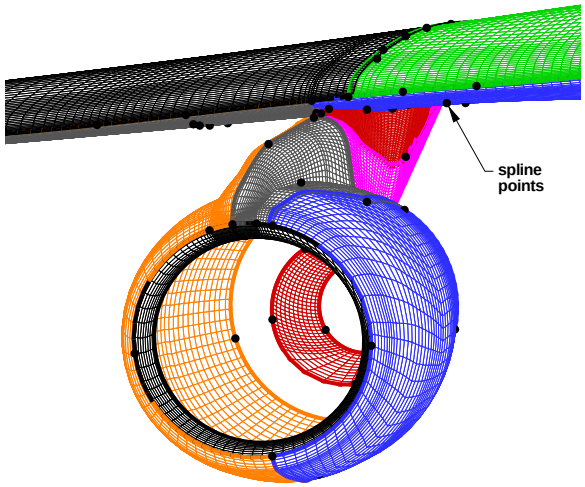

Figure 10 Lockheed L-1011 Nacelle/Pylon Multiblock Grid System

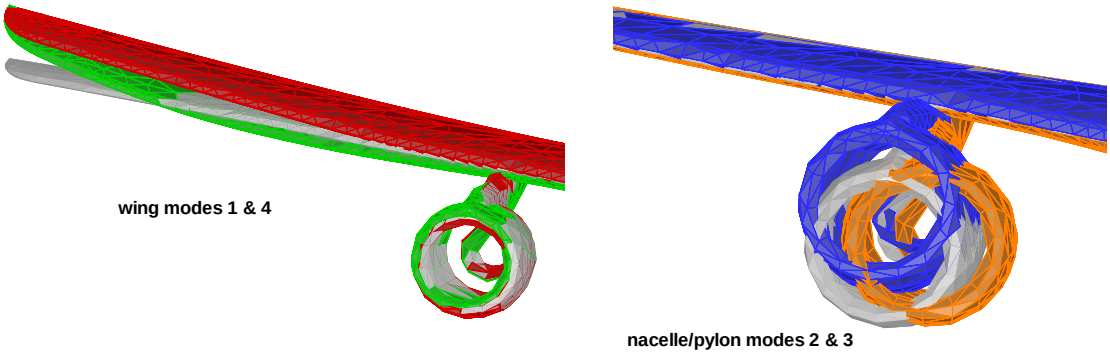

Figure 11 Lockheed L-1011 Structural Mode Shapes

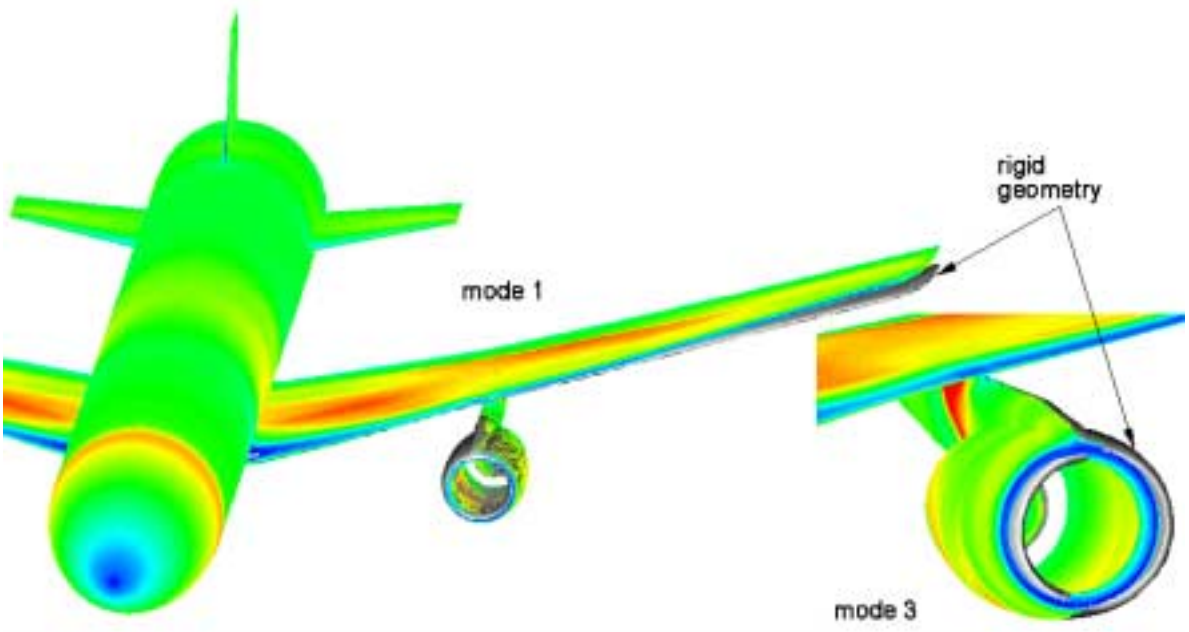

Figure 12 Lockheed L-1011 Forced Modal Motion Inviscid Aeroelastic Solution, M = 0.88, α = 0.0°